

DOCTORAL DISSERTATION

**TOMRAS: A Task Oriented Mobile Remote Access  
System for Desktop Applications**

Khaled Khankan  
kkhankan@it.uts.edu.au

Supervised by:  
Prof. Robert Steele  
robert.steele@usyd.edu.au

January 2009

Submitted in partial fulfilment of the requirements for the degree of  
Doctor of Philosophy in computer science.

© 2009 by Khaled Khankan  
This research was supported with ARC project LP0562624.

## **CERTIFICATE OF AUTHORSHIP/ORIGINALITY**

I certify that the work in this thesis has not previously been submitted for a degree nor has it been submitted as part of requirements for a degree except as fully acknowledged within the text.

I also certify that the thesis has been written by me. Any help that I have received in my research work and the preparation of the thesis itself has been acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

Signature of Candidate



Date 17/07/2009

---

# Acknowledgment

Many people deserve my thanks and appreciation for their contribution in helping me get to where I am today. I would like to express my sincere gratitude and great appreciation to my supervisor, Professor Robert Steele. His guidance and assistance have been of great value to me. His encouragement and support in tough times have been invaluable.

I also wish to thank Ms Sue Felix, for her help in editing and proofreading this thesis. Indeed, her dedication and support has made a difference.

My final, and most heartfelt thank-you goes to my wonderful mother, wife, and kids. Your patience over the years is unforgettable.

# Contents

<b>Chapter 1</b>	<b>Introduction.....</b>	<b>1</b>
1.1	Remote Access and Mobility.....	3
1.2	Motivation.....	4
1.3	Research Challenges.....	6
1.4	Advantages of Task-Oriented Mobile Remote Access.....	9
1.5	The Research Objectives and Scope.....	12
1.6	Thesis Significance and Contributions.....	12
1.7	The Research Methodology.....	15
1.8	Terms and Concepts.....	18
1.9	Structure of the Thesis.....	19
<b>Chapter 2</b>	<b>Background and Related Work.....</b>	<b>21</b>
2.1	Remote Access Computing: Introduction.....	23
2.2	Remote Access Computing Paradigms.....	24
2.2.1	Desktop-to-Desktop Remote Access.....	24
2.2.2	Mobile-to-Desktop Remote Access.....	24
2.3	Intelligent User Interface.....	25
2.3.1	User Interface Auto-generation.....	25
2.3.2	Mobile User Interface Auto-generation.....	27
2.4	Mobile User Interface and Remote Access.....	29
2.4.1	Text-Based User Interface and Remote Access.....	30
2.4.2	Graphics Based User Interface and Remote Access.....	32
2.4.3	Discussion.....	34
2.5	Multimodal User Interface.....	37
2.5.1	Multimodal Interaction in the Mobile Computing Domain.....	40
2.5.2	Multimodal User Interface and Mobile Remote Access.....	41
2.6	User Interface Description and Related Technologies.....	41
2.6.1	Abstract User Interface Description Markup Languages.....	42
2.6.2	Markup Languages for Describing the Multimodal User Interface.....	45
2.6.3	Discussion.....	48
2.7	Key Challenges and Issues with Mobile Remote Access.....	49
2.8	Conclusion.....	52

<b>Chapter 3</b>	<b>A Conceptual Model for Task-Oriented Mobile Remote Access .....</b>	<b>53</b>
3.1	Introduction.....	54
3.2	Terms and Concepts.....	56
3.3	Task-Oriented Modelling.....	59
3.4	TOMRAS Conceptual Model: Overview .....	59
3.5	The Task Recording Phase of the Model.....	64
3.5.1	Task User Interface Recognition and Refactoring .....	66
3.5.2	User Interaction Recording .....	67
3.5.3	Describing a Task.....	68
3.5.4	Tasks Registering .....	72
3.6	The Task Execution Phase of the Model .....	72
3.6.1	Task User Interface Adapting.....	72
3.6.2	Task Event Mapping .....	73
3.7	Extending the TOMRAS Model for Multimodal Task-Oriented User Interface .....	74
3.8	Conclusion .....	75
<b>Chapter 4</b>	<b>The TOMRAS Implementation Architecture for Task Recording.....</b>	<b>76</b>
4.1	Introduction.....	77
4.2	The Task Recording Architecture: An Overview .....	77
4.3	The Architecture Components .....	78
4.4	The Task Recording Mechanism .....	82
4.5	Describing the Task's User Interface.....	85
4.6	Conclusion .....	88
<b>Chapter 5</b>	<b>The TOMRAS Implementation Architecture for Task Execution .....</b>	<b>89</b>
5.1	Introduction.....	90
5.2	Overview of the Task Execution Architecture .....	90
5.3	The Architecture Components .....	93
5.3.1	The presentation tier.....	93
5.3.2	The middle tiers.....	96
5.3.3	The remote applications tier .....	97
5.4	The Task Execution Mechanism.....	98
5.5	Conclusion .....	103
<b>Chapter 6</b>	<b>TOMRAS Prototype .....</b>	<b>104</b>
6.1	Introduction.....	105
6.2	The Prototype Scope.....	106



6.3	The Prototype Implementation .....	108
6.3.1	Overview .....	108
6.3.2	The Mobile Thin Client and Events Manipulation.....	109
6.3.3	The Application Controller Implementation .....	112
6.3.4	Implementing the Application Automation Manager .....	114
6.4	A Sample Task for a Shipment Status Updating (SSU) .....	115
6.4.1	The SSU Task Recording .....	116
6.4.2	The SSU Task Execution .....	119
6.5	TOMRAS Prototyping Challenges and Limitations.....	122
6.6	Summary and Conclusion.....	122
<b>Chapter 7</b>	<b>Evaluation and Discussion .....</b>	<b>124</b>
7.1	Introduction.....	125
7.2	The Evaluation Methodology .....	126
7.2.1	The Predictive Models: An Overview.....	127
7.2.2	TOMRAS Evaluation and The Predictive Models.....	130
7.3	The Evaluation Configuration .....	131
7.3.1	The platforms .....	131
7.3.2	The benchmark remote access systems .....	131
7.3.3	The sample desktop applications.....	132
7.3.4	The sample tasks .....	135
7.4	TOMRAS Functionality Evaluation .....	136
7.4.1	Experiment 1: A Shipment Status Updating (SSU) Task.....	136
7.4.2	Experiment 2: A Task for Adding Patient Examination Results....	137
7.4.3	Experiment 3: A Medical Record Updating Task Via <i>FileMed</i> .....	140
7.5	Analytically Evaluating the Effectiveness of the Task-Oriented User Interface .....	141
7.5.1	Case Study 1: Predicting the Time Required for the SSU Task....	142
7.5.1.1	Using the Keystroke Level Model (KLM) .....	142
7.5.1.2	Using Fitts' law-based model .....	146
7.5.2	Case Study 2: Predicting the Time Required for the PMS Task ...	151
7.5.2.1	Using the KLM: .....	151
7.5.2.2	Using Fitts' law-based model .....	154
7.5.3	Case Study 3: Predicting the Time Required for the <i>FileMed</i> Task .....	158
7.5.3.1	Using the KLM.....	158
7.5.3.2	Using Fitts' law-based model .....	162
7.6	Results Analysis and Discussion .....	170
7.7	Issues and Limitations .....	173
7.8	Extending the TOMRAS Prototype.....	174
7.9	Conclusion .....	175

<b>Chapter 8</b>	<b>Conclusions.....</b>	<b>177</b>
8.1	Summary.....	178
8.1.1	Mobile Remote Access Computing.....	179
8.1.2	The Research Methodology.....	180
8.1.3	TOMRAS Model and Architecture.....	181
8.1.4	TOMRAS Evaluation and Experimental Results.....	181
8.2	Contributions.....	182
8.3	Limitations and Future Directions.....	183
8.4	Concluding Remarks.....	185
<b>Appendix A</b>	<b>The TOMRAS Prototype Source Code.....</b>	<b>188</b>
A.1	The Mobile Thin Client.....	188
A.1.1	The ‘MobileAppManager’ class:.....	188
A.1.2	The ‘Communicator’ Class:.....	190
A.1.3	Helper Classes or Utilities:.....	192
A.2	The Application Controller:.....	202
A.3	The Application Automation Manager:.....	204
A.3.1	The ‘AppAutomationServer’ class:.....	204
A.3.2	The ‘AppMonitor’ Class:.....	211
<b>References</b>	<b>.....</b>	<b>215</b>

# List of Figures

Figure 1: Screenshot of a desktop application for shipment management .....	1
Figure 2: Screenshots of a desktop application while being remotely accessed via LogMeIn.....	1
Figure 3: A design research methodology . Adapted from:.....	1
Figure 4: A high level view of Remote Access Computing .....	23
Figure 5: A High level diagram for task-oriented model of mobile remote access ..	1
Figure 6: The task recording processes.....	65
Figure 7: Mapping of task's desktop user interface into mobile user interface .....	1
Figure 8: An implementation architecture for the TOMRAS task recording phase ..	1
Figure 9: A sequence diagram for the task recording architecture .....	1
Figure 10: An implementation architecture for the TOMRAS task execution phase .....	92
Figure 11: A sequence diagram for the task execution architecture .....	1
Figure 12: A class diagram for the prototyped mobile thin client .....	1
Figure 13: A screenshot for a refactored UI of the (SSU) task displayed on a PDA. ....	1
Figure 14: An instance of generated identification maps for the SSU task user interface control identifiers. ....	1
Figure 15: Screenshot of a desktop application for shipment management .....	1
Figure 16: Screenshot for the patient management system ( <i>PMS</i> ) desktop application. ....	1
Figure 17: A medical record view in the FileMed desktop application.....	1
Figure 18: A screen for adding examination results in PMS. ....	1
Figure 19: A screenshot for the generated GUI of the patient diagnoses update task. ....	1
Figure 20: Refactored user interface for the FileMed task. ....	1
Figure 21: The predicted time for the SSU task according to the KLM.....	1
Figure 22: The predicted time for performing the SSU task according to the Fitts' law based model.....	1
Figure 23: The predicted time for the <i>PMS</i> task according to the KLM .....	1
Figure 24: The predicted time for performing the PMS task according to the Fitts' law based model.....	1
Figure 25: The predicted time for the FileMed task according to the KLM.....	1
Figure 26: The predicted time for the FileMed task according to Fitts' Law based model.....	1



# List of Tables

Table 1. A comparison between graphics-based and text-based.....36

Table 2. User interface abstract description entities .....69

Table 3. Attribute of the user interface elements .....69

Table 4. Windows messages representing user interaction.....81

Table 5. Excluded user interface elements and reasons.....117

Table 6 A standard set of approximate times.....129

Table 7. Applying the KLM to the SSU task via *LogMeIn* .....143

Table 8. Applying the KLM to the SSU task via *RDM+*.....144

Table 9. Applying the KLM to the SSU task via TOMRAS .....145

Table 10. Applying Fitts' Model to the SSU task via *LogMeIn*.....146

Table 11. Applying Fitts' Model to the SSU task via *RDM+*.....147

Table 12. Applying Fitts' Model to the SSU task via TOMRAS .....149

Table 13. Applying the KLM to a *PMS* task via *LogMeIn* .....151

Table 14. Applying the KLM to a *PMS* via *RDM+*.....152

Table 15. Applying the KLM to a *PMS* via TOMRAS .....153

Table 16. Applying Fitts' Model to a *PMS* task via *LogMeIn* .....154

Table 17. Applying Fitts' Model to a *PMS* task via *RDM+* .....155

Table 18. Applying Fitts' Model to a *PMS* task via TOMRAS .....156

Table 19. Applying the KLM to a *FileMed* task via *LogMeIn* .....158

Table 20. Applying the KLM to a *FileMed* task via *RDM+*.....159

Table 21. Applying the KLM to a *FileMed* task via TOMRAS .....161

Table 22. Applying Fitts' Model to a *FileMed* task via *LogMeIn*.....162

Table 23. Applying Fitts' Model to a *FileMed* task via *RDM+* .....165

Table 24. Applying Fitts' Model to a *FileMed* task via TOMRAS .....167

Table 25: A summary of the TOMRAS usability testing results.....171

# List of Listings

Listing 1: A registered task entry .....1

Listing 2: An XML-based notation for a task’s user interface description.....1

Listing 3: A pseudo code for the mobile application manager .....1

Listing 4: A pseudo-code for the core of the mobile thin client application .....1

Listing 5: A pseudo-code for the ‘KeyDown’ and ‘Click’ event handlers .....1

Listing 6: A pseudo-code for the Application Automation Manager .....1

Listing 7: The identified user interface of the main window of the ‘Shipment  
Management’ desktop application .....1

Listing 8: The refactored user interface description of the SSU task .....1

Listing 9: A code extract of the identified user interface of the patient diagnoses  
update dialog .....1

Listing 10: A code extract of a refactored UI description of the patient diagnoses  
update task.....1

Listing 11: A code extract of a refactored user interface description of the medical  
record update task .....1

# Abstract

Mobile remote access to desktop applications is a potential enabler to improving the productivity and convenience of individuals and businesses. There is an increasing research interest in developing mobile remote access solutions for desktop applications. The developed proposals, however, are challenged by the hardware limitations of most mobile devices, such as the small display size. These limitations have a direct impact on the way existing desktop applications are presented on mobile devices.

This thesis focuses on developing new ways of achieving effective mobile remote access to existing desktop applications. A conceptual model and implementation architecture for a task-oriented mobile remote access system (TOMRAS) have been introduced. The TOMRAS model adopts a user interface refactoring approach to generate task-oriented user interfaces for existing desktop applications without re-developing or modifying these applications.

TOMRAS has a number of novel aspects, including, inferring user interface and behaviour knowledge from existing applications and transparently exposing the functionalities of existing desktop applications to be remotely accessible via a wide spectrum of mobile devices and platforms without redeveloping these desktop applications. The TOMRAS strategy of decoupling the generated mobile task's user interface from the functionality of existing applications also allows for a possible enriching of the mobile task's user interface with multimodal interaction capabilities.

The thesis describes the TOMRAS conceptual model, and a potential implementation architecture for this model. The proposed implementation architecture articulates the intrinsic building blocks for mobile remote access

solutions that adopt the TOMRAS model. The limitations of how widely and generically the model and techniques can be applied are also detailed in the thesis. Furthermore, a prototype that validates the feasibility of the TOMRAS implementation architecture is provided, and an evaluation of the effectiveness of the task-oriented approach is presented.